UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/921,868	08/03/2001	Vikas Agarwal	JP920010088US1	7137

7590 02/03/2006

McGinn & Gibb, PLLC
2568-A Riva Road
Suite 304
Annapolis, MD 28211

EXAMINER	
PATEL, ASHOKKUMAR B	
ART UNIT	**PAPER NUMBER**
2154	

DATE MAILED: 02/03/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/921,868	AGARWAL ET AL.	
	Examiner	Art Unit	
	Ashok B. Patel	2154	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *26 November 2005*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-20* is/are pending in the application.

4a) Of the above claim(s) *7,10-12,14,17 and 18* is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-6,8,9,13,15,16,19 and 20* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-20 are subject to examination. Claims 7, 10-12, 14, 17 and 18 have been cancelled.

Response to Arguments

2. Applicant's arguments filed 11/26/2005 have been fully considered but they are not persuasive for the following reasons:

Applicant's argument:

"These features are neither taught nor suggested in Abrams or Microsoft. In particular, Abrams does not teach a process to identify failures on its networked machines (i.e., to detect faulty machines), let alone using this identifying information to determine the number of application instances of one or more resource class components that should be changed/fluctuated. Furthermore, Abrams does not teach that faulty machines (i.e., machines comprising failures) do not receive allocations resources. Abrams briefly describes on page 6, paragraph [00641, "Some of the advantages provided by the on-demand method and system 140 include: protection during peak loads, in one embodiment, with guaranteed application response time SLA; global reach with application provider control of distributed web presence; freedom to grow aggressively including elastic web-processing infrastructure on demand; no capital investment with costs based. on the amount of capacity used; supporting substantially any application on substantially any platform to preserve application provider's current application investment; and higher reliability because the system provides superior response time and automatically routes around failures." However, there is no teaching of what constitutes "failures" and

whether this routing around failures means that the failures reside on the machines themselves or the applications, or the resource components, and if this means that the failures are fixed or ignored or removed from the system. Also, there is no teaching in Abrams of identifying failures within a particular amount of time (i.e., within a time constraint) (i.e., before a timing out process occurs), which the Applicants' claimed invention provides."

"The virtual server in Abrams is a set of physical servers serving an application for one customer. Whereas, the virtual server provided by the Applicants' invention is defined as a multiered application which can include multiple instances of each tier (i.e., resource classes)."

Examiner's answer:

Abrams et al. claims priority of the provisional application 60/232,052, a copy of which was provided to the Applicant along with the previous non-final office Action and prior to the argued features of Abrams et el. that are now added as amendments.

This provisional application provides the description of failures as well as the servers as follows:

Please note the definition of "Compute Node (CN)" at 2.2.2. and "Failure Modes" provided on pages 14 and 15 as shown below.

2 Terminology

2.1 Site Sites represent customers such as pets.com.

2.1.1 MainSite This is the Site that is run and maintained by the customer. In an EPN serviced site, this will handle the "Buy" button

2.1.2 DeveloperSite This is the Site that is run and maintained by the customer as a part of their development methodology.

2.1.3 OutsourcedSite This is the portion of the Site that is handled by the EPN.

2.1.4 ApplicationInstance This is the Site application entity that is runnable at an EdgePoint. Each Site can have multiple ApplicationInstances (AI) at each EdgePoint to service a request. A URL to AI relationship is many-many.

2.2 EdgePoint(EP) EdgePoint is a collection of machines that run the customers site. The EdgePoint runs inside a locked cabinet or rack at a data center. EdgePoints are managed by Ejasent and in one embodiment Release 1 will not be backed up.

2.2.1 Dispatch Node(DN) Dispatch Node is the local dispatcher within an EdgePoint that dispatches incoming requests. If the request cannot be routed to a running ApplicationInstance, the Dispatch Node schedules an ApplicationInstance to complete an incoming request. Typical requests are http requests. However, non-http requests such has rtsp protocols can also be serviced

2.2.2 Compute Node(CN) Compute Node is the component within an EdgePoint that runs the components of an ApplicationInstance. Typically, these are application tiers such as a webserver connected to a middle-tier and a database.

2.2.3 Admin Node(AN) Admin Node is the component within an EdgePoint that runs the administrative components of an EdgePoint. The configuration database, deployer components, the data synchronization components and the monitoring components are run here.

2.2.4 Edge Processing Network Edge Processing Network is the engineering entity that, using multiple EdgePoints, creates a geographically dispersed computing fabric to support on-demand scalability, and lower response time at a substantially lower cost.

2.2 GlobalDispatcher(GP) GlobalDispatcher (or DistributedDispatcher) has the primary function of connecting a request to the best EP that can service the request. The requests that come in are called "unresolved" and the connection "resolves" the request. The "best server" determination is based on network latency and server loading metrics.

2.3 Conduit(CP) Conduit is the primary way a MainSite communicates with the EdgePoints. It abstracts the distributed nature of the EPN and allows the customer (Site) to update, manage and view their data and applications without being burdened by the location and load of the actual EdgePoints

2.4 Deployment Center(DP) Deployment Center acts as the hub that collects data, policies and applications from the conduit and sends (and receives) them to the EdgePoints (spokes). Deployment Center maintains application/data versions.

2.4.1 Deployment Deployment is the method of capturing application state (initial and updates), policies and testing methods from the Conduit machine, moving it to the Deployment Center and then to the EPs. The policies will include deployment and execution policies. Application state includes the actual application data/binaries and the method to create the snapshots.

2.5 Data synchronization Data synchronization is the method used to send user created or app generated data (html, jpg, gif, jsp, catalog, clickstream and log) from the Conduit to the EPs (via DP), or from the EPs to the Conduit. If the data is sent out of Conduit, it is called fan-out and if it comes back to the Conduit, it is called fan-in. Data synchronization is achieved with messaging middleware.

2.6 Cleaving Cleaving is process of dividing a Site into EPN handled requests and MainSite handled requests. Cleaving is done using the Studio on the Conduit by Ejasent Professional Services. Cleaving will require the pages in the MainSite to be modified to allow redirection to the EPN global dispatcher. *Cleaving creates the OutsourcedSite at the Conduit.*

2.7 Isolation Isolation is defined as the method by which an AI for a Site is protected from another AI belonging to another Site, while still sharing the same hardware, software, network

5 Failure modes

The following are the different failure modes in the EPN.

5.1 Down AN or CN or SAN or local network

Since the DN is up, it can convert resolved requests to unresolved requests and redirect to the GD that will redirect to another EP. Session data on the down CN will be temporarily (or permanently) lost.

5.2 Down DN or external EP network

If the Dispatcher Node (DN) is down at an EP, and if the global dispatcher (GP) is aware the EdgePoint is down, then it will not dispatch any requests to that EP.

If the DN is down and the client is bound to that specific EdgePoint (cookie/URL) then we can:

- Dispatch to the EdgePoint anyway and expose the client to the failure. The user will need to restart the request with an unresolved name to get out of this problem.
- Use another DN within the EP if a redundant DN configuration is used.

5.3 Down AI (memory or disk state)

If an ApplicationInstance goes down, DN will stop routing to that AI and route to a new AI or re-route back to global dispatcher. Session information in the AI will be temporarily (or permanently) lost.

When an AI goes down, the AI has to be brought back up and then re-snapshotted. If the AI has a database program instance, instance recovery of the database will recover the database (as in Oracle). Restoring a previous snapshot will not work for AIs with program instances with updated databases (Oracle or other updated databases) but may work for other AIs. The Studio captures the method of restoring an AI in the conduit.

5.4 Down DP

Deployment Center is redundant. A new deployment center can be used for deployment. However, any data synchronization in transit will be temporarily lost. They are not permanently lost because the data synch mechanism is end-end persistent and the DP is simply a hop in the transport.

5.5 Down GP (global dispatcher)

GP is redundant. Hence, another GP can be used. The redirection to a GP is via normal DNS mechanisms, which support primary and secondary hosts.

5.6 Down Conduit

In some cases Conduit is redundant. In that case, the new Conduit can be used for deployment if both conduits had the same deployment state. Data synchronization will be temporarily halted, which means some selections may be lost. We will allow the Site admin the option to force GP to route all requests to MainSite.

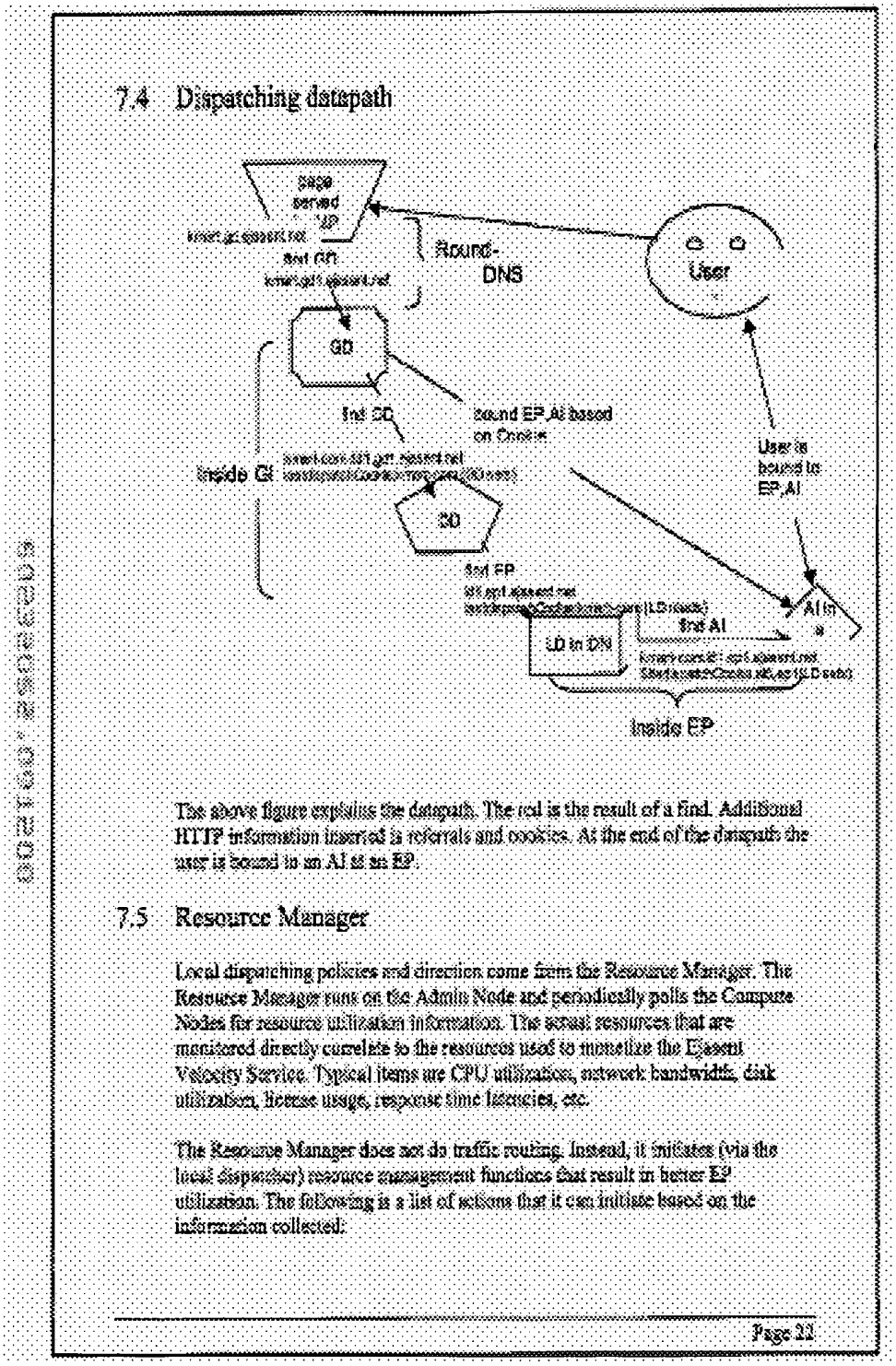
Applicant's argument:

"Abrams uses 'appshot as a technique to increase and decrease the application capacity in response to changing load. Conversely, the Applicants' invention provides a computational load to control the allocation of resources in a fine-grained manner."

"Conversely, the Applicants' invention allocates resources to customers based on current load and past usage history (i.e., changed number of application instances of one or more resource class components)."

Examiner's response:

Also please note the functions of the resource manager on page 22 and 23 as shown below.



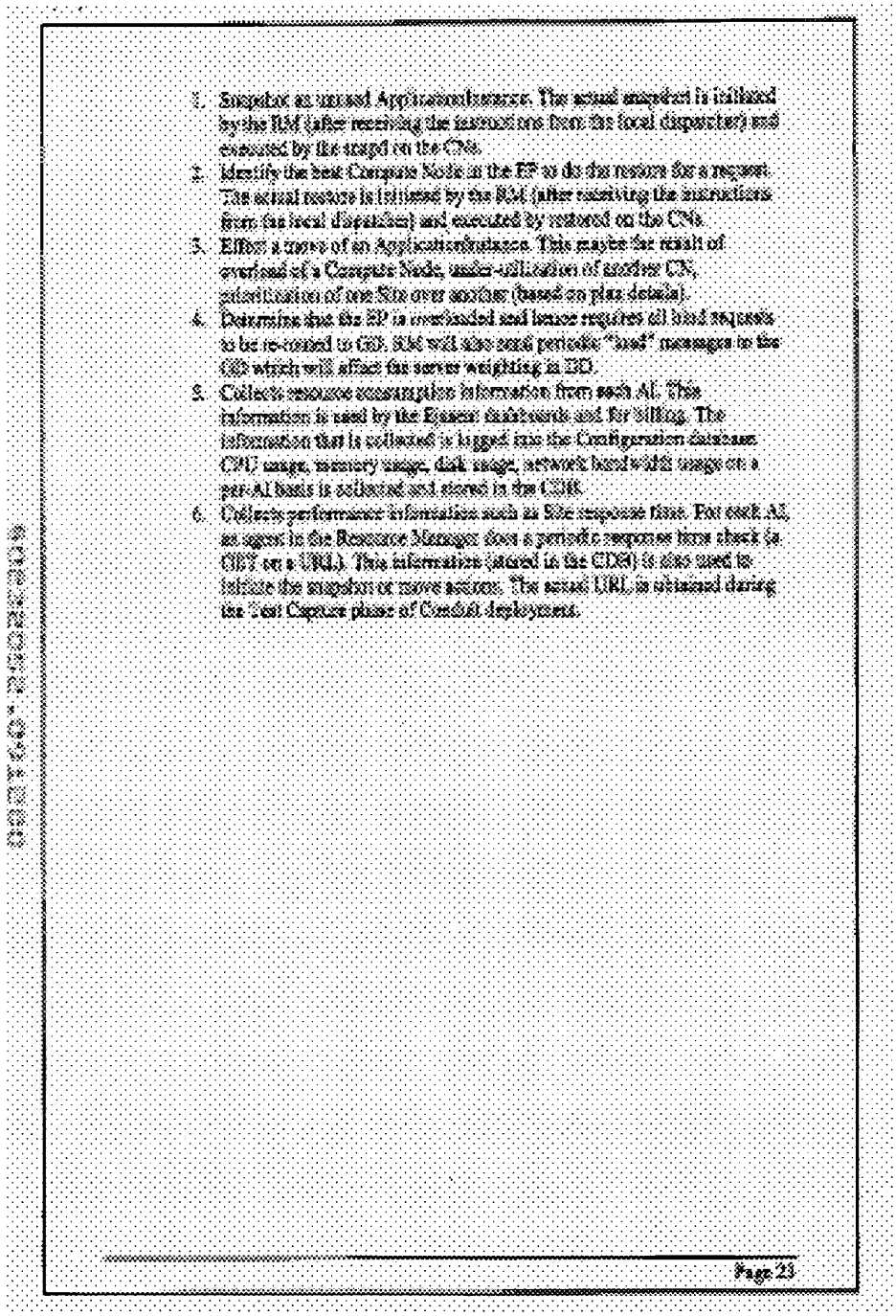
7.4 Dispatching datapath

The above figure explains the datapath. The red is the result of a find. Additional HTTP information inserted is referrals and cookies. At the end of the datapath the user is bound to an AI at an EP.

7.5 Resource Manager

Local dispatching policies and direction come from the Resource Manager. The Resource Manager runs on the Admin Node and periodically polls the Compute Nodes for resource utilization information. The actual resources that are monitored directly correlate to the resources used to monetize the Ejasent Velocity Service. Typical items are CPU utilization, network bandwidth, disk utilization, license usage, response time latencies, etc.

The Resource Manager does not do traffic routing. Instead, it initiates (via the local dispatcher) resource management functions that result in better EP utilization. The following is a list of actions that it can initiate based on the information collected:



1. Snapshot an unused Application instance. The actual snapshot is initiated by the RM (after receiving the instructions from the local dispatcher) and executed by the stagd on the CNs.
2. Identify the best Compute Node in the EP to do the restore for a request. The actual restore is initiated by the RM (after receiving the instructions from the local dispatcher) and executed by restored on the CNs.
3. Effect a move of an Application instance. This may be the result of overload of a Compute Node, under-utilization of another CN, prioritization of one Site over another (based on plan details).
4. Determine that the EP is overloaded and hence requires all load requests to be re-routed to GD. RM will also send periodic "load" messages to the GD which will affect the server weighting in GD.
5. Collects resource consumption information from each AI. This information is used by the Report database and for billing. The information that is collected is logged into the Configuration database. CPU usage, memory usage, disk usage, network bandwidth usage on a per-AI basis is collected and stored in the CDB.
6. Collects performance information such as Site response time. For each AI, an agent in the Resource Manager does a periodic response time check (a GET on a URL). This information (stored in the CDB) is also used to initiate the snapshot or move actions. The actual URL is obtained during the Test Capture phase of Conduit deployment.

Page 23

3. Claim rejections of claims 1, 13, 15, 16, 19 and 20 are withdrawn based on the response provided by the Applicant.

Claim Rejections - 35 USC § 102

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
> *(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.*

5. Claims 1-3, 5, 6, 8, 9, 13, 15, 16, 19 and 20 are rejected under 35 U.S.C. 102(e) as being anticipated by Abrams et al. (hereinafter Abrams) (US 2002/0166117 A1)

Referring to claim 1,

Abrams teaches a method of providing access for a plurality of application-level users to an application comprising a plurality of resource class components comprising tiered layers of web servers, commerce servers, and database servers collectively executing on multiple networked machines (Abstract) , the method comprising of:

receiving an incoming flow of requests from application-level users to use an application and components of said application (Abstract);

providing, for each of the application-level users, respective sets of one or more application instances of each resource class component for the application on one or

more machines, to service the incoming requests from respective application-level users to use the application (Abstract, page 2, para.[0021]);

directing each of the incoming requests to a particular application instance of an appropriate resource class component (page 6, para.[0067]);

monitoring, for each of the application-level users, the number of request serviced by the application instances of the resource class components of the application (Abstract);

identifying, within a time frame constraint, failures on any of said multiple networked machines; (page 6, para.[0064])

changing the number of application instances of one or more resource class components in response to the monitored number of requests for each resource class component and based on machines comprising failures (Abstract, page 2, para.[0021], (page 6, para.[0064]));

maintaining a record of the current rate of requests received from respective application-level users based on the monitored number of serviced requests (page 2, para.[0021]); and

collectively and automatically allocating fractions of different resource class components to a particular application-level user in response to the changed number of application instances of one or more resource class components by using a computational load of each request imposing on said application, wherein said computational load corresponds to a number of requests allocated for each resource instance wherein said machines comprising failures are prevented from receiving

allocations of resources. (page 5, para.[0062], page 6, para.[0067], page 2, para.[0019],[0020] and [0021], (page 2, para.[0021]), page 11, para.[0091]).

Referring to claim 2,

Abrams teaches the method as claimed in claim 1, further comprising:

directing each of the incoming requests from respective application-level users to a particular application instance of an appropriate resource class component from a respective set of one or more application instances of each resource class component, said particular application instance being identified as the least loaded of the application instances of the appropriate resource class component from that respective set. (page 6, para.[0067])

Referring to claim 3,

Abrams teaches the method as claimed in claim 1, wherein the step of providing application instances of each resource class component further comprises: initiating one or more application instance of one or more resource class on a plurality of machines to service incoming requests to use the application (page 6, para.[0067],[0068]); and terminating one or more application instances of each resource class on a plurality of machines to service incoming requests to use the application (page 2, para.[0021]).

Referring to claim 5,

Abrams teaches the method as claimed in claim 1, further comprising: maintaining a record of service obligations to respective application-level users. (page 6, para.[0064], page 14, para. [0125])

Referring to claim 6,

Abrams teaches the method as claimed in claim 5, further comprising changing, for each of the application-level users, the number of application instances of each resource class component in response to the monitored number of requests for each resource class component, wherein the service obligations to respective application-level users are at least met. (page 6, para.[0064], page 14, para. [0125], page 8, para.[0078]).

Referring to claim 8,

Abrams teaches the method as claimed in claim 1, wherein said step of changing the number of application instances of said one or more resource classes in (i) at least partly based upon said recorded current rate of requests received from respective application-level users, (page 8, para.[0074]) and (ii) at least partly based on predetermined information that correlates changes in request rates with charges in the corresponding number of application instances of said one or more resource classes required to service said request rates.(page 6, para.[0068], page 8, 0078])

Referring to claim 9,

Abrams teaches the method as claimed in claim 1, wherein one or more of the application-level users are organizations, and the requests are generated by individuals associated with the respective organization. (page 5, para. [0059])

Referring to claim 13,

Abrams teaches the method of providing access for a plurality of application-level

users to an application comprising a plurality of resource class components comprising tiered layers of web servers, commerce servers, and database servers collectively executing on multiple networked machines (Abstract), the method comprising steps of:

receiving an incoming flow of requests from application-level users to use an application and components of said application (Abstract);

providing, for each of the application-level users, respective sets of one or more application instances of each resource class component for the application on one or more machines, to service the incoming requests from the application-level users to use the application (Abstract, page 2, para.[0021]);

monitoring, for each of the application-level users, the resources currently available and resources currently consumed by the requests serviced by application instances of the resource class components of the application (Abstract);

identifying, within a time frame constraint, failures on any of said multiple networked machines; (page 6, para.[0064])

maintaining a record of resources currently available to respective application-level users; and a record of resources currently consumed by respective application-level users; both records of said resources being maintained in respect of each of the one or more application instances of each resource class components (page 6, para.[0067]);

adjusting the respective numbers of said one or more application instances of each component (Abstract, page 2, para.[0021]); and

collectively and automatically allocating fractions of different resource class components to a particular application-level user in response to a fluctuating number of application instances of one or more resource class components by using a computational load of each request imposing on said application, wherein said computational load corresponds to a number of requests allocated for each resource instance wherein said machines comprising failures are prevented from receiving allocations of resources. (page 5, para.[0062], page 6, para.[0067], page 2, para.[0019],[0020] and [0021], (page 2, para.[0021]), page 11, para.[0091]).

wherein said application instances of each resource class component are adjusted for each application-level user based (i) at least partly on said records of resources currently available and currently consumed by respective application-level users (page 8, para.[0074]). and (ii) at least partly on predetermined information that estimates the number of each resource class components required to service requests for said application instances of the resource class components (page 6, para.[0068], page 8, 0078]), and (iii) at least partly on machines comprising failures. (page 6, para.[0064])

Referring to claim 15,

Claim 15 is a claim to a system that carries out the steps of method of claim 1. Therefore claim 15 is rejected for the reasons set forth for claim 1.

Referring to claim 16,

Claim 16 is a claim to a computer software program, recorded on a medium and capable of execution of steps of method of claim 1. Therefore claim 16 is rejected for the reasons set forth for claim 1.

Referring to claim 19,

Claim 19 is a claim to a system that carries out the steps of method of claim 13. Therefore claim 19 is rejected for the reasons set forth for claim 13.

Referring to claim 20,

Claim 20 is a claim to a computer software program, recorded on a medium and capable of execution of steps of method of claim 13. Therefore claim 20 is rejected for the reasons set forth for claim 13.

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

7. Claim 4 is rejected under 35 U.S.C. 103(a) as being unpatentable over Abrams et al. (hereinafter Abrams) (US 2002/0166117 A1) in view of Microsoft Computer Dictionary (hereinafter Microsoft) Published in 1997.

Referring to claim 4,

Keeping in mind the teachings of Abrams, although Abrams teaches at para.[0125], page 14, "Execution policies relate to user-level SLAs and priorities for execution.", Abrams fails to specifically teach , wherein requests from application-level users to use

the application are stored in a queue for execution by a particular application instance of the appropriate resource class on a first-in-first-out basis.

Microsoft teaches " a method of processing a queue, in which they were removed in the same order in which they were added – the first in is the first out.

Therefore, it would have been obvious to one having ordinary skill in the art at the time of invention was made to prioritize the execution of the requests of Abrams per Microsoft such that same user level SLAs are executed in a first-in-first-out basis.

Conclusion

Examiner's note: Examiner has cited particular columns and line numbers in the references as applied to the claims above for the convenience of the applicant. Although the specified citations are representative of the teachings of the art and are applied to the specific limitations within the individual claim, other passages and figures may apply as well. It is respectfully requested from the applicant in preparing responses, to fully consider the references in entirety as potentially teaching all or part of the claimed invention, as well as the context of the passage as taught by the prior art or disclosed by the Examiner.

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the

shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ashok B. Patel whose telephone number is (571) 272-3972. The examiner can normally be reached on 8:00am-5:00pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John A. Follansbee can be reached on (571) 272-3964. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Abp
***

JOHN FOLLANSBEE
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100